สัมฤทธิ์ แสนธิ

SOFT MAIL

สร้างและใช้งานกับ FOXPRO ตอนจบ

ก้าวไปอีกขั้นกับการสร้างเมนูใน FOXPRO ด้วยคำสั่งขั้นสูงสุด

ฉบับที่แล้วผู้เขียนได้กล่าวถึงการสร้างและใช้งานเมนูโดยใช้คำสั่งเบื้องต้นไปเรียบร้อยแล้ว ฉบับนี้ก็จะขอนำเสนอคำสั่งขั้นสูงในการสร้างและใช้งานเมนูกันต่อ ในฟอกซ์โปรจะทำการเขียนโปรแกรมสำหรับสร้างและใช้งานเมนูเก็บไว้ในรูทีน (routine) เรียบร้อยแล้ว เมื่อมีการใช้งานคำสั่งเพียงแต่ศึกษาวิธีการใช้งานของเมนูเท่านั้นก็สามารถจะออกแบบเมนูสวย ๆ ไว้ใช้งานได้อย่างง่ายดาย

คำสั่งขั้นสูง

คำสั่งขั้นสูงสำหรับสร้างและใช้งานเมนูจะมีความยุ่งยากในการใช้งานมากกว่าคำสั่งเบื้องต้น แต่จะมีรายละเอียดและความสามารถของเมนูมากกว่า เหมาะที่จะใช้สำหรับเขียนโปรแกรมเมนูเป็นระบบ (system menu) การสร้างเมนูจะทำการสร้างไว้ในหน่วยความจำก่อนแล้วจึงสั่งแอคตีฟเพื่อใช้งานทีหลัง ดังนั้นจึงใช้เนื้อที่ของหน่วยความจำค่อนข้างมาก เครื่องคอมพิวเตอร์ที่ใช้จึงต้องมีหน่วยความจำอย่างน้อย 640 กิโลไบต์ขึ้นไป สำหรับรายละเอียดของคำสั่งมีดังนี้

คำสั่ง DEFINE MENU

```
DEFINE MENU <menu name> [MESSAGE <expC>]
```

คำสั่ง DEFINE MENU ใช้สำหรับสร้างเมนูบาร์ (menu bar) ไว้ในหน่วยความจำ โดยจะใช้คำสั่ง DEFINE PAD สำหรับกำหนดข้อความของหัวข้อเมนูบาร์ที่จะสร้างขึ้น ดังนั้นการสร้างเมนูบาร์จึงต้องใช้สองคำสั่งร่วมกัน คำสั่ง DEFINE MENU จะสร้างเมนูบาร์ไว้ในหน่วยความจำยังไม่สามารถใช้งานได้จะต้องใช้คำสั่ง ACTIVATE MENU สั่งแอคตีฟเมนูที่สร้างขึ้นจึงจะสามารถใช้งานได้ ส่วนการใช้งานเมนูจะใช้คีย์บอร์ดหรือเมาส์เหมือนการใช้งานเมนูบาร์ที่สร้างจากคำสั่งเบื้องต้นที่กล่าวมาแล้ว

<menu name> คือชื่อของเมนูบาร์ที่จะสร้างขึ้นเก็บไว้ในหน่วยความจำ

MESSAGE <expC> เป็น

ออปชันที่ใช้สำหรับแสดงคำอธิบายหรือหมายเหตุบนจอสกรีนที่ตำแหน่งกึ่งกลางของบรรทัดที่เซตไว้ในคำสั่ง SET MESSAGE แต่ถ้าไม่ได้สั่งเซตไว้จะแสดงไว้ที่บรรทัดล่างสุดของจอสกรีน <expC> คือข้อความที่ใช้เป็นคำอธิบายหรือหมายเหตุ ตัวอย่างการใช้งานคำสั่งนี้ให้ดูจากรูปที่ 1

```
SET TALK OFF
CLEAR
SET MESSAGE TO 1
DEFINE MENU mainmenu MESSAGE "SOFT WARE"
SET MESSAGE TO 5
DEFINE PAD mache OF mainmenu PROMPT "\<Machine";
AT 3,00, MESSAGE "Machine Language"
DEFINE PAD assem OF mainmenu PROMPT "\<Assembly";
AT 3,20, MESSAGE "Assembly Language"
DEFINE PAD high  OF mainmenu PROMPT "\<High livel";
AT 3,40, MESSAGE "High Livel language"
DEFINE PAD quit OF mainmenu PROMPT "\<Quit";
AT 3,60, MESSAGE "Quit To FOXPRO"
ON SELECTION PAD mache OF mainmenu DO tmenu;
WITH MENU() , PAD()
ON SELECTION PAD assem OF mainmenu DO tmenu;
WITH MENU() , PAD()
ON SELECTION PAD high  OF mainmenu DO tmenu;
WITH MENU() , PAD()
ON SELECTION PAD quit  OF mainmenu DO tmenu;
WITH MENU() , PAD()
ACTIVATE MENU mainmenu
WAIT "End Program" TIMEOUT 5
PROCEDURE tmenu
PARAMEter mmenu,mpad
HIDE MENU mainmenu SAVE
@ 4,0 SAY "Select menu Pad Of " + mmenu + "is" + mpad
WAIT
IF mpad ="QUIT"
   DEACTIVATE MENU
ENDIF
CLEAR
RETURN
```

รูปที่ 1 แสดงตัวอย่างโปรแกรมและคำสั่งในการใช้งาน

คำสั่ง DEFINE PAD

```
DEFINE PAD <pad name> OF <menu name> PROMPT <expC1> [AT <row>, <col>] [MESSAGE <expC2>]
```

เป็นคำสั่งที่ใช้สำหรับสร้างข้อความของหัวข้อเมนูบาร์ซึ่งจะใช้ร่วมกับคำสั่ง DEFINE MENU ที่กล่าวมาแล้ว คำสั่ง DEFINE PAD หนึ่งคำสั่งจะสร้างหัวข้อของเมนูบาร์ได้เพียงหัวข้อเดียว ถ้าต้องการสร้างหัวข้อเมนูบาร์หลาย ๆ หัวข้อจะต้องใช้คำสั่ง DEFINE PAD หลาย ๆ คำสั่งตามจำนวนของหัวข้อเมนูบาร์ที่ต้องการสร้างขึ้น

<pad name> คือชื่อของหัวข้อเมนูบาร์ที่สร้างขึ้น ถ้ามีหลายหัวข้อชื่อของหัวข้อเมนูบาร์จะต้องไม่ซ้ำกัน

<menu name> คือชื่อของเมนูบาร์ที่ต้องการสร้างจากคำสั่ง DEFINE MENU ที่กล่าวมาแล้ว

PROMPT <expC1> ใช้สำหรับกำหนดข้อความของหัวข้อเมนูบาร์ที่ต้องการแสดงบนจอสกรีน ซึ่งจะใช้สัญลักษณ์ "\<" นำหน้าตัวอักษรที่ต้องการสร้างเป็นคีย์ด่วน (Hot Key)

AT <row>, <col> เป็นออปชันที่ใช้สำหรับกำหนดตำแหน่งที่จะให้แสดงเมนูบาร์บนจอสกรีน โดยค่าของ <row> จะมีค่าตั้งแต่ 0 ถึง 24 และค่าของ <col> จะมีค่าตั้งแต่ 0 ถึง 79 ถ้าไม่ระบุออปชันนี้ในคำสั่งของเมนูบาร์จะแสดงไว้ที่ตำแหน่ง (0, 0) ของจอสกรีนและจะแสดงหัวข้อเมนูหัวข้อใหม่ ในตำแหน่งที่ถัดไปทางด้านขวาของหัวข้อเมนูเดิม

MESSAGE <expC2> ถ้าระบุออปชันนี้ในคำสั่งด้วยจะเป็นออปชันที่ใช้สำหรับคำอธิบายหรือหมายเหตุเกี่ยวกับหัวข้อของเมนูบาร์ <expC2> คือข้อความที่ใช้เป็นคำอธิบายหรือหมายเหตุ ซึ่งจะแสดงไว้ที่กึ่งกลางของบรรทัดบนจอสกรีนที่สั่งเซตไว้ในคำสั่ง SET MESSAGE แต่ถ้าไม่ได้สั่งเซตไว้จะแสดงไว้ที่บรรทัดล่างสุดของจอสกรีน ตัวอย่างการใช้งานคำสั่งให้ดูจากรูปที่ 1

คำสั่ง ACTIVATE MENU

```
ACTIVATE MENU <menu name> [PAD <pad name>]
```

คำสั่งนี้จะใช้ร่วมกับคำสั่ง DEFINE MENU และคำสั่ง DEFINE PAD สำหรับแอคตีฟเมนูบาร์เพื่อใช้งานหลังจากสร้างเมนูบาร์ด้วยคำสั่งทั้งสองที่กล่าวมาเรียบร้อยแล้ว คำสั่ง ACTIVATE MENU จะแสดงเมนูบาร์บนจอสกรีนหรือในวินโดว์ที่กำหนดและสามารถใช้งานเมนูบาร์นั้นได้ทันทีโดยใช้คีย์บอร์ดหรือเมาส์

<menu name> คือชื่อของเมนูบาร์ที่สร้างไว้ในหน่วยความจำจากคำสั่ง DEFINE MENU

PAD <pad name> เป็นออปชันที่ใช้สำหรับกำหนดไฮไลต์ที่หัวข้อของเมนูบาร์ที่ต้องการหลังจากเมนูบาร์แอคตีฟ แต่ถ้าไม่ระบุออปชันนี้ไฮไลต์จะแสดงที่หัวข้อของเมนูบาร์หัวข้อแรกในคำสั่ง DEFINE PAD สำหรับ

<pad name> คือชื่อของหัวข้อเมนูบาร์ที่สร้างขึ้นจากคำสั่ง DEFINE PAD ที่กล่าวมาแล้ว ตัวอย่างการใช้งานได้แสดงในรูปที่ 1

ฟังก์ชัน MENU

MENU ( )

ค่าของฟังก์ชันนี้จะเป็นข้อความ (String) เป็นฟังก์ชันที่ใช้สำหรับเก็บชื่อของเมนูบาร์ที่สร้างจากคำสั่ง DEFINE MENU ในขณะที่เมนูบาร์นั้นแอคตีฟอยู่ ดังนั้นถ้ามีการสร้างเมนูบาร์หลาย ๆ เมนูจึงสามารถใช้ฟังก์ชัน MENU หาชื่อของเมนูบาร์ที่แอคตีฟได้ สำหรับชื่อของเมนูบาร์ที่ได้จากฟังก์ชันนี้จะถูกแปลงเป็นตัวพิมพ์ใหญ่ทั้งหมดและถ้าไม่มีเมนูบาร์แอคตีฟค่าของฟังก์ชันจะเป็นช่องว่าง ตัวอย่างในการใช้งานให้ดูจากรูปที่ 1

ฟังก์ชัน PAD

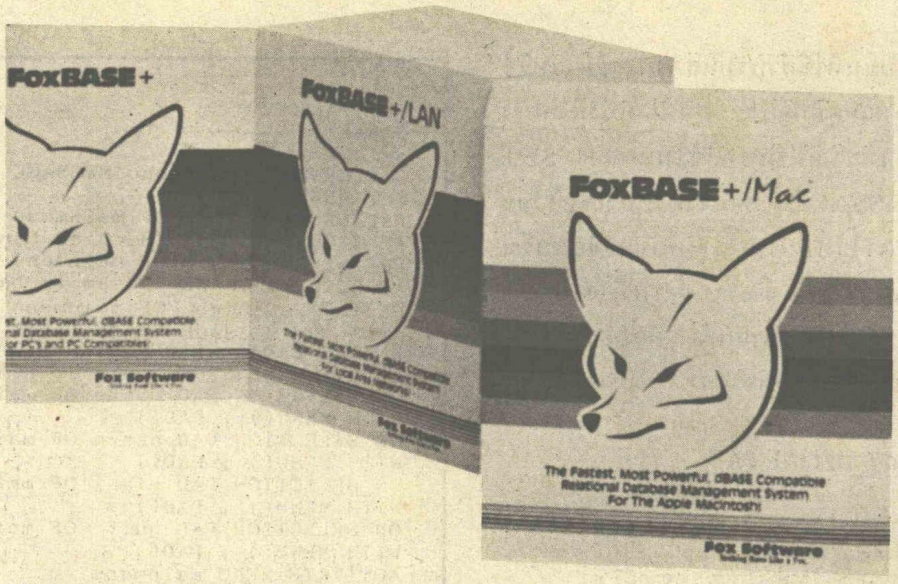
PAD ( )

ค่าของฟังก์ชันนี้จะเป็นข้อความ (String) ใช้สำหรับเก็บชื่อหัวข้อของเมนูบาร์ (ข้อความใน <pad name> จากคำสั่ง DEFINE PAD) เมื่อเมนูบาร์แอคตีฟและถูกเรียกใช้งานโดยคีย์บอร์ดหรือเมาส์ข้อความที่ได้จากฟังก์ชันจะถูกแปลงเป็นตัวอักษรตัวพิมพ์ใหญ่ทั้งหมด แต่ถ้าเมนูบาร์ยังไม่ถูกแอคตีฟ ค่าของฟังก์ชันที่ได้จะเป็นช่องว่าง สำหรับตัวอย่างการใช้งานให้ดูจากรูปที่ 1

คำสั่ง ON SELECTION PAD

```
ON SELECTION PAD <pad name> OF
<menu name> [<command>]
```

เป็นคำสั่งที่ใช้สำหรับเช็คค่าในการเรียกใช้งานเมนูบาร์ในหัวข้อต่าง ๆ ซึ่งจะใช้หนึ่งคำสั่งต่อหนึ่งหัวข้อของเมนูบาร์ ถ้ามีการเรียกใช้หัวข้อของเมนูตรงตามที่กำหนดไว้ใน <pad name> คำสั่งที่ระบุใน <command> จะถูกเอ็กซีคิวต์ (execute) (ถ้าระบุคำสั่งที่จะเอ็กซีคิวต์ไว้ในคำสั่งด้วย)



<pad name> คือชื่อของหัวข้อเมนูบาร์ที่กำหนดไว้ในคำสั่ง DEFINE PAD

<menu name> คือชื่อของเมนูบาร์ที่กำหนดไว้ในคำสั่ง DEFINE MENU

<command> คือคำสั่งในฟอกซ์โปรที่ต้องการให้เอ็กซีคิวต์เมื่อหัวข้อของเมนูบาร์ที่ระบุใน <pad name> ถูกเรียกใช้งาน แต่ถ้าไม่ระบุ <command> เมนูบาร์ก็จะถูกแอคตีฟต่อไป สำหรับตัวอย่างการใช้งานให้ดูจากรูปที่ 1

คำสั่ง DEACTIVATE MENU

DEACTIVATE MENU

เป็นคำสั่งที่ใช้สำหรับยกเลิกการแอคตีฟของเมนูบาร์ซึ่งจะลบเมนูบาร์บนจอสกรีนและในวินโดว์ออกด้วย แต่ยังคงเก็บรายละเอียดของเมนูบาร์ไว้ในหน่วยความจำ จึงสามารถเรียกใช้งานเมนูบาร์ได้จาก ACTIVATE MENU หรือแสดงเมนูบาร์ได้จากคำสั่ง SHOW MENU สำหรับตัวอย่างการใช้งานให้ดูจากรูปที่ 1

คำสั่ง SHOW MENU

```
SHOW MENU <menu name1> [,<menu
name2>...] : ALL [PAD <pad name>]
[SAVE]
```

เป็นคำสั่งที่ใช้สำหรับแสดงเมนูบาร์ที่สร้างไว้ในหน่วยความจำจากคำสั่ง DEFINE MENU และคำสั่ง DEFINE PAD บนจอสกรีนหรือวินโดว์ คำสั่งนี้จะแสดงเมนูบาร์เท่านั้นไม่สามารถใช้งานเมนูได้

<menu name> คือชื่อของเมนูบาร์ที่สร้างขึ้นจากคำสั่ง DEFINE MENU ถ้ามีการสร้างเมนูบาร์หลาย ๆ เมนูและต้องการให้แสดงบนจอสกรีนหรือวินโดว์พร้อมกันให้ระบุชื่อของเมนูที่ต้องการลงในคำสั่งตามลำดับที่ต้องการให้แสดง (<menu name1>, <menu name2>,...) หรือระบุเป็น All Option แทน ถ้าต้องการให้แสดงเมนูบาร์ทุกเมนูที่สร้างไว้ในหน่วยความจำ

PAD <pad name> เป็นออปชันที่ใช้สำหรับกำหนดให้แสดงไฮไลต์ที่หัวข้อของเมนูบาร์ ซึ่งจะมีผลเมื่อใช้

คำสั่ง ACTIVATE MENU ก่อนใช้คำสั่ง SHOW MENU เท่านั้น

SAVE ถ้าระบุออปชันนี้ในคำสั่งจะทำให้เมนูบาร์แสดงค้างไว้บนจอสกรีนหรือวินโดว์หลังจากใช้คำสั่ง HIDE MENU, คำสั่ง DEACTIVATE MENU, คำสั่ง CLEAR MENU หรือคำสั่ง RELEASE MENU สำหรับตัวอย่างการใช้งานให้ดูจากรูปที่ 2

คำสั่ง HIDE MENU

```
HIDE MENU <menu name1> [,<menu name2>...] : ALL [SAVE]
```

เป็นคำสั่งที่ตรงกันข้ามกับคำสั่ง SHOW MENU ที่กล่าวมาแล้ว เป็นคำสั่งที่ใช้สำหรับลบเมนูบาร์ที่แสดงบนจอสกรีนหรือวินโดว์ออก แต่จะไม่ลบรายละเอียดของเมนูบาร์ที่เก็บในหน่วยความจำ ดังนั้นถ้าเรียกใช้คำสั่ง ACTIVATE MENU หรือคำสั่ง SHOW MENU จึงสามารถใช้งานหรือแสดงเมนูบาร์ได้อีกครั้งหนึ่ง สำหรับรายละเอียดการใช้งานออปชันต่าง ๆ ในคำสั่งนี้จะเหมือนกับในคำสั่ง SHOW MENU ดังนั้นจึงไม่ขอกล่าวอีก สำหรับตัวอย่างการใช้งานให้ดูจากรูปที่ 2

คำสั่ง CLEAR MENU และคำสั่ง RELEASE MENU

```
CLEAR MENU RELEASE MENU [<menu list>]
```

ทั้งสองคำสั่งนี้จะใช้สำหรับลบเมนูบาร์ออกจากหน่วยความจำ จอสกรีนและวินโดว์ทำให้ไม่สามารถเรียกเมนูนั้นกลับมาแสดงหรือใช้งานได้อีก สำหรับคำสั่ง RELEASE MENU ถ้าไม่ระบุชื่อของเมนูบาร์ที่จะลบไว้ใน <menu list> เมนูบาร์ทุกเมนูในหน่วยความจำจะถูกลบทิ้ง สำหรับตัวอย่างการใช้งานให้ดูจากรูปที่ 2

คำสั่ง DEFINE POPUP

```
DEFINE POPUP <popup name> FORM <row1>, <col1> [TO <row2>, <col2>] [PROMPT FIELD <field> : PROMPT FILES [LIKE <skel>] : PROMPT STRUCTURE] [MESSAGE <expC>] [SHADOW] [COLOR <standard> [,<enhanced>] : COLOR <expN>
```

คำสั่งนี้มีออปชันในการใช้งานค่อนข้างมาก แต่เป็นการเพิ่มความสามารถของเมนูที่สร้างให้ใช้งานได้มากขึ้น คำสั่ง DEFINE POPUP ใช้สำหรับสร้างเมนูทางแนวตั้งหรือป๊อปอัพเมนู (เหมือนคำสั่ง @...PROMPT แต่มีรายละเอียดในการใช้งานเมนูมากกว่า) ไว้ในหน่วยความจำ

การสร้างป๊อปอัพเมนูสามารถสร้างได้หลายวิธี คือ สร้างจากข้อมูลที่ได้จากคำสั่ง DEFINE BAR, สร้างจากข้อมูลในแฟ้มดาต้าเบสที่เปิดใช้งาน, สร้างจาก Structure ของดาต้าเบสที่เปิดใช้งานและสร้างจากรายชื่อของไฟล์ในแผ่นดิสเกตต์ เป็นต้น คำสั่ง DEFINE POPUP คำสั่งเดียวจะยังไม่สามารถใช้งานได้ จะต้องใช้งานร่วมกับคำสั่งอื่น ๆ อีกเช่น คำสั่ง DEFINE BAR, คำสั่ง ACTIVATE POPUP, ฟังก์ชัน BAR, ฟังก์ชัน PROMPT, ฟังก์ชัน POPUP ฯลฯ

<popup name> คือชื่อของป๊อปอัพเมนูที่จะสร้างขึ้นแล้วเก็บไว้ในหน่วยความจำ

FROM <row1>, <col1> คือตำแหน่งมุมบนด้านซ้ายของกรอบเมนูป๊อปอัพที่ต้องการให้แสดงบนจอสกรีนหรือในวินโดว์ที่สร้างขึ้น

TO <row2>, <col2> คือตำแหน่งมุมล่างด้านขวาของกรอบป๊อปอัพเมนูที่ต้องการให้แสดงบนจอสกรีนหรือในวินโดว์ที่สร้างขึ้น ถ้าไม่ระบุออปชันนี้ในคำสั่งขนาดของป๊อปอัพเมนูจะถูกกำหนดขึ้นเองโดยอัตโนมัติและถ้าขนาดของป๊อปอัพเมนูยาวเกินไปจนเกินขอบเขตของจอสกรีนหรือวินโดว์ทำให้แสดงข้อความของหัวข้อป๊อปอัพเมนูไม่พอ ส่วนที่เหลือจะถูกซ่อนไว้ในกรอบของป๊อปอัพเมนูซึ่งจะสามารถใช้ Arrow Key หรือคีย์ PgUp หรือ PgDn บนคีย์บอร์ดหรือใช้เมาส์เลื่อน

```
SET TALK OFF
CLEAR
DEFINE MENU mainmenu
DEFINE PAD invoices OF mainmenu PROMPT "\<Invoices";
AT 06,00
DEFINE PAD customers OF mainmenu PROMPT "\<Customers";
AT 06,20
ON SELECT PAD invoices OF mainmenu DO tmenu
ACTIVATE MENU mainmenu
WAIT
RELEASE MENU mainmenu
ACTIVATE MENU mainmenu
*************************
PROCEDURE tmenu
HIDE MENU mainmenu
WAIT "*HIHE MENU DEMO*" TIMEOUT 2
SHOW MENU mainmenu PAD customers SAVE
WAIT "*SHOW MENU DEMO*" TIMEOUT 2
CLEAR
RETURN
```

รูปที่ 2 แสดงตัวอย่างโปรแกรมและคำสั่งในการใช้งาน

ไฮไลต์หาเพื่อเรียกใช้งานได้

PROMT FIELD <field> เป็นออปชันที่ใช้สำหรับนำเอาข้อมูลในฟิลด์ (field) (ซึ่งระบุโดย <field> ของดาต้าเบสที่เปิดใช้งานมาเป็นข้อความของหัวข้อป๊อปอัพเมนูที่สร้างขึ้น โดยข้อมูลในเรคอร์ดที่ 1 จะเป็นข้อความของหัวข้อเมนูที่ 1 ข้อมูลในเรคอร์ดที่ 2 จะเป็นหัวข้อเมนูที่ 2 ลำดับอย่างนี้เรื่อยไป ตัวอย่างการใช้งาน เช่น ในแฟ้มข้อมูลชื่อ CUSTOMER.DBF มีข้อมูลแต่ละ Record ใน Field ชื่อ LIST_FIELD ดังรูปที่ 3

RECORD	LIST_FIELD
#1	BOOK
#2	COMPUTER
#3	TABLE
#4	PEN
#5	QUIT

รูปที่ 3
แสดงแฟ้มข้อมูลชื่อ CUSTOMER.DBF

ตัวอย่างการเขียนโปรแกรมสร้างป๊อปอัพเมนูจากข้อมูลในแฟ้ม CUSTOMER.DBF เขียนได้ดังรูปที่ 4

```
SET TALK OFF
CLEAR
USE customer
DEFINE POPUP customers  FROM 5,5 PROMPT;
FIELD list_field
ACTIVATE POPUP customers
```

รูปที่ 4 ตัวอย่างการเขียนโปรแกรมสร้างป๊อปอัพเมนู

PROMPT FILES [LIKE <skel>] เป็นออปชันที่ใช้สร้างป๊อปอัพเมนูจากรายชื่อของไฟล์ในแผ่นดิสเกตต์ทุกไฟล์ในดิสค์ไดรฟ์หรือฮาร์ดดิสค์ที่ติดตั้งไว้ในเครื่องคอมพิวเตอร์ แต่ถ้าระบุ LIKE ออปชันจะสามารถระบุชื่อไดรฟ์ ชื่อสารบบย่อยและชื่อไฟล์ (สามารถใช้สัญลักษณ์ "*" และสัญลักษณ์ "?" แทนชื่อและนามสกุลไฟล์ได้) ที่ต้องการใน <skel> เพื่อนำมาสร้างเป็นป๊อปอัพเมนูได้ ตัวอย่างการเขียนโปรแกรมใช้งาน เช่น สมมติว่าต้องการนำเอาเฉพาะไฟล์ที่มีนามสกุล .DBF จากสารบบย่อยข้อมูลในไดรฟ์ A มาสร้างป๊อปอัพเมนู สามารถเขียนโปรแกรมได้ดังรูปที่ 5

```
SET TALK OFF
CLEAR
DEFINE POPUP list_files FROM PROMPT FILES;
LIKE A:\DATA\*.DBF
ACTIVATE MENU list_files
```

รูปที่ 5 ตัวอย่างการเขียนโปรแกรมใช้งานเฉพาะไฟล์ที่มีนามสกุล .DBF

PROMPT STRUCTURE เป็นออปชันที่สร้างสำหรับป๊อปอัพเมนูจากโครงสร้างของดาต้าเบสที่เปิดใช้งาน โดยนำเอาชื่อของฟิลด์มาเป็นข้อความของหัวข้อป๊อปอัพเมนูซึ่งจะทำให้สามารถเขียนโปรแกรมใช้งานแฟ้มดาต้าเบสได้คล่องตัวขึ้น ตัวอย่างการใช้งานดังรูปที่ 6

```
SET TALK OFF
CLEAR
DEFINE POPUP list_field FROM 5,5 PROMPT STRUCTURE
ACTIVATE MENU list_field
```

รูปที่ 6 ตัวอย่างการเขียนโปรแกรมโดยนำชื่อฟิลด์มาเป็นข้อความของหัวข้อป๊อปอัพเมนู

MESSAGE <expC> ถ้าระบุออปชันนี้ในคำสั่ง จะสามารถแสดงคำอธิบายหรือหมายเหตุเกี่ยวกับหัวข้อของป๊อปอัพเมนูบนจอสกรีนที่ตำแหน่งซึ่งเซตไว้ในคำสั่ง SET MESSAGE หรือที่บรรทัดล่างสุดของจอสกรีนถ้าไม่ได้สั่งเซตตำแหน่งที่จะแสดงไว้

COLOR <standard>, <enhanced> | COLOR SCHEME <expN> เป็นออปชันที่ใช้สำหรับกำหนดสีของกรอบและตัวอักษรที่ใช้ในป๊อปอัพเมนู สำหรับตัวอย่างการสร้างป๊อปอัพเมนูจากข้อมูลในคำสั่ง DEFINE BAR แสดงได้ดังรูปที่ 7

คำสั่ง DEFINE BAR

```
DEFINE BAR <expN> OF <popup
name> PROMPT <expC1> [MESSAGE
<expC2> [SKIP [FOR <expL>]]]
```

เป็นคำสั่งที่ใช้สำหรับสร้างข้อความของหัวข้อป๊อปอัพเมนูไว้ในหน่วยความจำ ซึ่งจะต้องใช้ร่วมกับคำสั่ง DEFINE POPUP คำสั่งนี้หนึ่งคำสั่งจะสร้างหัวข้อป๊อปอัพเมนูได้เพียงหนึ่งหัวข้อเท่านั้น ดังนั้นถ้าต้องการสร้างหัวข้อของป๊อปอัพเมนูหลาย ๆ หัวข้อ จะต้องใช้คำสั่งนี้สร้างเท่ากับจำนวนหัวข้อของป๊อปอัพเมนูที่ต้องการสร้าง

<expN> คือลำดับที่ของหัวข้อป๊อปอัพเมนูที่สร้างขึ้น

```
SET TALK OFF
CLEAR
DEFINE POPUP invpop FROM 02,01
DEFINE BAR 1 OF invopo PROMPT "\<The invoice stack"
DEFINE BAR 2 OF invopo PROMPT "\<Print one or more invoices";
SKIP
DEFINE BAR 3 OF invopo PROMPT "\<Show me what my saie are"
DEFINE BAR 4 OF invopo PROMPT "s\<Cane database" SKIP FOR .T.
DEFINE BAR 5 OF invopo PROMPT "\<Quit"
ON SELECTION POPUP invpop DO tmenu WITH POPUP(),PROMPT(),BAR()
ACTIVATE POPUP invpop
PROCEDURE tmenu
PARAMETER mpop,mprompt,mbar
HIDE POPUP invpop
@ 5,0 SAY "Selected prompt of popup" + mbar + mpop + "is";
+ prompt
WAIT
IF mbar = 5
   DEACTIVATE POPUP
ENDIF
CLEAR
RETURN
```

รูปที่ 7 ตัวอย่างการใช้งานในโปรแกรมป๊อปอัพเมนู

<popup name> คือชื่อของป๊อปอัพเมนูที่สร้างจากคำสั่ง DEFINE POPUP

<expC1> คือข้อความของหัวข้อป๊อปอัพเมนูที่จะสร้างขึ้น ซึ่งสามารถใช้สัญลักษณ์ "\<" นำหน้าตัวอักษรที่ต้องการสร้างเป็นคีย์ด่วน (Hot Key) หรือใช้สัญลักษณ์ "\-" สำหรับขีดเส้นคั่นระหว่างหัวข้อของป๊อปอัพเมนูที่ต้องการ

MESSAGE <expC2> เป็นออปชันที่ใช้สำหรับแสดงคำอธิบายหรือหมายเหตุของหัวข้อป๊อปอัพเมนูบนจอสกรีนหรือวินโดว์ที่ตำแหน่งซึ่งได้เซตไว้ในคำสั่ง SET MESSAGE หรือที่บรรทัดสุดท้ายของจอสกรีนกรณีที่ไม่ได้สั่งเซตตำแหน่งไว้

SKIP [FOR<expL>] เป็นออปชันที่ใช้สำหรับยกเลิกการใช้งานหัวข้อของป๊อปอัพเมนูอย่างถาวรหรือชั่วคราว โดยระบุ Skip option หรือ Skip for Option ตามลำดับ การยกเลิกชั่วคราวจะใช้เงื่อนไขใน <expL> เป็นตัวเช็ค ถ้าเงื่อนไขเป็นจริง (True = .T.) จะเป็นการยกเลิกการใช้งานหัวข้อของป๊อปอัพเมนูแต่ถ้าเงื่อนไขเป็นเท็จ (False = .F.) จะทำให้สามารถใช้งานหัวข้อของป๊อปอัพเมนูได้ตามปกติ ตัวอย่างการใช้งานให้ดูจากรูปที่ 7

คำสั่ง ON SELECTION POPUP

```
ON SELECTION POPUP <popup name>
: ALL [<command>]
```

เป็นคำสั่งที่ใช้สำหรับกำหนดการเอ็กซีคิวต์คำสั่งต่อไป ถ้าเรียกใช้งานป๊อปอัพเมนูตรงตามที่กำหนดไว้ใน <popup name> ของคำสั่งนี้ คำสั่งที่ระบุใน <command> จะถูกเอ็กซีคิวต์ทันที

<popup name> | ALL เป็นชื่อของป๊อปอัพเมนูที่กำหนดไว้ในคำสั่ง DEFINE POPUP หรือถ้าต้องการเช็คการเรียกใช้งานของป๊อปอัพเมนูพร้อมกันหลาย ๆ เมนูจะใช้ All Option แทนชื่อของป๊อปอัพเมนูเหล่านั้น

<command> คือคำสั่งในฟอกซ์โปรที่ต้องการให้เอ็กซีคิวต์เมื่อหัวข้อในป๊อปอัพเมนูถูกเรียกใช้งาน สำหรับตัวอย่างการใช้งานให้ดูจากรูปที่ 7

ฟังก์ชัน POPUP

POPUP ( )

ใช้สำหรับเก็บชื่อของป๊อปอัพเมนูที่แอคตีฟและถูกใช้งาน ถ้าไม่มีป๊อปอัพเมนูแอคตีฟค่าของฟังก์ชันจะเป็นช่องว่าง ตัวอย่างการใช้งานคำสั่งนี้ให้ดูได้จากรูปที่ 7

ฟังก์ชัน PROMPT

PROMPT ( )

ใช้สำหรับเก็บข้อความในหัวข้อของป๊อปอัพเมนูที่กำหนดไว้ใน PROMPT ออปชันของคำสั่ง DEFINE BAR เมื่อป๊อปอัพเมนูถูกสั่งแอคตีฟและหัวข้อของป๊อปอัพเมนูนั้นถูกเรียกใช้งาน แต่ถ้าไม่มีป๊อปอัพเมนูแอคตีฟต่อของฟังก์ชันจะเป็นช่องว่าง ถ้าป๊อปอัพเมนูถูกสร้างจาก Database, Structure หรือ File ค่าของฟังก์ชัน PROMPT ที่ได้จะเป็นข้อมูลจาก Database, Structure หรือ File แทน สำหรับตัวอย่างการใช้งานคำสั่งนี้ให้ดูได้จากรูปที่ 7

ฟังก์ชัน BAR

BAR ( )

เป็นฟังก์ชันที่ใช้สำหรับหาลำดับที่ของหัวข้อป๊อปอัพเมนู (หาค่า <expN> จากคำสั่ง DEFINE BAR) เมื่อป๊อปอัพเมนูถูกแอคตีฟและหัวข้อใดหัวข้อหนึ่งถูกเรียกใช้งาน แต่ถ้าคีย์ Esc บนคีย์บอร์ดถูกกด ค่าของฟังก์ชันจะมีค่าเท่ากับศูนย์ ซึ่งหมายถึงการยกเลิกการใช้งานป๊อปอัพเมนูนั้น สำหรับตัวอย่างการใช้งานคำสั่งนี้ให้ดูได้จากรูปที่ 7

คำสั่ง ACTIVATE POPUP

ACTIVATE POPUP <popup name>

คำสั่งนี้จะใช้ได้เมื่อมีการสร้างป๊อปอัพเมนูจากคำสั่ง DEFINE POPUP และคำสั่ง DEFINE BAR เรียบร้อยแล้ว เป็นคำสั่งที่ใช้สำหรับแอคตีฟและแสดงป๊อปอัพเมนูที่สร้างไว้ในหน่วยความจำ บนจอสกรีนหรือในวินโดว์ที่กำหนด ส่วนการยกเลิกการใช้งานเมนูนั้นจะใช้คำสั่ง DEACTIVATE POPUP หรือกดคีย์ Esc สำหรับตัวอย่างการใช้งานคำสั่งนี้ให้ดูได้จากรูปที่ 7

คำสั่ง DEACTIVATE POPUP

DEACTIVATE POPUP

คำสั่งนี้จะมีหน้าที่ตรงกันข้ามกับคำสั่ง ACTIVATE POPUP ใช้สำหรับทำหน้าที่ยกเลิกการใช้งานป๊อปอัพเมนูทุกเมนูที่แอคตีฟอยู่และลบป๊อปอัพเมนูนั้นออกจากจอสกรีนหรือวินโดว์แต่ยังคงเก็บรายละเอียดของเมนูไว้ในหน่วยความจำ สามารถที่จะแสดงได้อีกครั้งด้วยคำสั่ง ACTIVATE POPUP หรือคำสั่ง SHOW POPUP การใช้งานคำสั่ง DEACTIVATE POPUP นี้จะใช้ได้เฉพาะการเขียนโปรแกรมเท่านั้น จะไม่มีผลถ้าสั่งให้ทำงานผ่านทาง Command Window สำหรับตัวอย่างการใช้งานคำสั่งนี้ให้ดูได้จากรูปที่ 7

คำสั่ง SHOW POPUP

SHOW POPUP <popup name1> [,<popup name2>...] : ALL

คำสั่งนี้จะใช้สำหรับแสดงป๊อปอัพเมนูไว้ (ซึ่งสร้างไว้ในหน่วยความจำจากคำสั่ง DEFINE POPUP และคำสั่ง DEFINE BAR) บนจอสกรีนหรือในวินโดว์ที่กำหนด แต่จะไม่สามารถใช้งานเมนูได้

```
SET TALK OFF
CLEAR
DEFINE POPUP inpop
DEFINE BAR 1 OF inpop PROMPT "\<Then invoice stick"
DEFFIN BAR 2 OF inpop PROMPT "\<Print one"
DEFINE BAR 3 OF inpop PROMPT "\<List my"
DEFINE BAR 4 OF inpop PROMPT "\<Show invoice"
DEFINE BAR 5 OF inpop PROMPT "\<Quit"
SHOW POPUP inpop
WAIT
HIDE POPUP inpop
WAIT
SHOW POPUP ALL SAVE
WAIT
HIDE POPUP inpop
WAIT
SHOW POPUP inpop
WAIT
RELEASE POPUP inpop
WAIT
SHOW POPUP inpop
```

รูปที่ 8 แสดงตัวอย่างโปรแกรมและวิธีการใช้งานในคำสั่งต่าง ๆ

<popup name1> [,<popup name2>...]¦ ALL เป็นชื่อของป๊อปอัพเมนูที่ต้องการให้แสดง ถ้าต้องการให้แสดงพร้อมกันหลาย ๆ เมนูให้ระบุชื่อตามลำดับที่จะให้เมนูแสดงก่อนหลังคั่นด้วยเครื่องหมาย "," หรือจะใช้ ALL ออปชันแทนการแสดงป๊อปอัพเมนูทั้งหมดก็ได้

SAVE เป็นออปชันที่ใช้สำหรับให้แสดงป๊อปอัพเมนูบนจอสกรีนหรือในวินโดว์ค้างไว้เมื่อมีการยกเลิกการใช้งานหรือลบเมนูด้วยคำสั่ง DEACTIVATE POPUP, คำสั่ง HIDE POPUP หรือคำสั่ง RELEASE POPUP สำหรับตัวอย่างการใช้งานคำสั่งนี้ให้ดูได้จากรูปที่ 8

คำสั่ง HIDE POPUP

HIDE POPUP <popup name> [,<popup name>...] [ALL [SAVE]

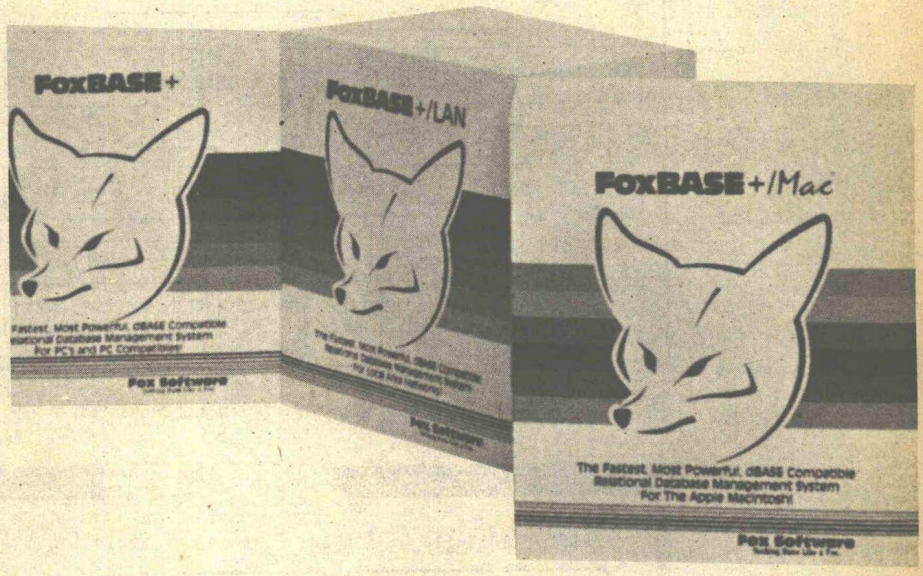
คำสั่งนี้จะตรงกันข้ามกับคำสั่ง SHOW POPUP คือใช้สำหรับลบป๊อปอัพเมนูออกจากจอสกรีนหรือในวินโดว์ แต่ป๊อปอัพเมนูนั้นจะยังคงเก็บไว้ในหน่วยความจำ สามารถเรียกใช้งานได้อีกเมื่อใช้คำสั่ง ACTIVATE POPUP หรือคำสั่ง SHOW POPUP รายละเอียดของการใช้งานของออปชันต่างๆ ในคำสั่งจะเหมือนกับคำสั่ง SHOW POPUP ซึ่งได้กล่าวไปแล้วจึงไม่ขอนำมากล่าวอีก ส่วนตัวอย่างการใช้งานนั้นให้ดูได้จากรูปที่ 8

คำสั่ง CLEAR POPUP และคำสั่ง RELEASE POPUP

CLEAR POPUP RELEASE POPUP <popup list>

ทั้งสองคำสั่งนี้จะใช้สำหรับลบป๊อปอัพเมนูออกจากหน่วยความจำ จอสกรีนและวินโดว์ ทำให้ไม่สามารถเรียกเมนูนั้นกลับมาแสดงหรือใช้งานได้อีก สำหรับคำสั่ง RELEASE MENU ถ้าไม่ระบุชื่อของป๊อปอัพเมนูที่จะลบไว้ใน <popup list> ป๊อปอัพเมนูทุกเมนูในหน่วยความจำจะถูกลบทิ้ง สำหรับตัวอย่างการใช้งานคำสั่งนี้ให้ดูได้จากรูปที่ 8

การสร้างและใช้งาน SYSTEM MENU



การสร้างซิสเต็มส์เมนูโดยใช้คำสั่งขั้นสูงนั้นค่อนข้างจะยุ่งยากกว่าการใช้คำสั่งเบื้องต้น แต่ซิสเต็มส์เมนูที่สร้างขึ้นจะมีรายละเอียดในการใช้งานมากกว่า คำสั่งที่ใช้ในการสร้างและใช้งานจะเป็นคำสั่งที่กล่าวมาแล้วนำมารวมกัน เช่น คำสั่ง DEFINE MENU, คำสั่ง DEFINE PAD, คำสั่ง DEFINE POPUP, คำสั่ง DEFINE BAR, คำสั่ง ACTIVATE MENU, คำสั่ง ACTIVATE POPUP, คำสั่ง ON SELECTION PAD, คำสั่ง ON SELECTION POPUP ฯลฯ นอกจากนี้ยังมีเพิ่มอีกหนึ่งคำสั่งคือคำสั่ง ON PAD ซึ่งจะขอกล่าวถึงรายละเอียดในลำดับต่อไปนี้

คำสั่ง ON PAD

```
ON PAD <pad name> OF <menu name> [ACTIVATE POPUP <popup name>]
```

คำสั่งนี้จะมีการใช้งานคล้ายคำสั่ง ON SELECTION PAD ต่างกันตรงที่คำสั่งนี้จะต้องแอคตีฟเมนูบาร์และป๊อปอัพเมนูให้สามารถใช้งานร่วมกัน ดังนั้นการใช้งานคำสั่งนี้จึงต้องมีการสร้างเมนูบาร์และป๊อปอัพเมนูไว้ในหน่วยความจำก่อน (โดยใช้คำสั่ง DEFINE MENU, คำสั่ง DEFINE PAD, คำสั่ง DEFINE POPUP และคำสั่ง DEFINE BAR ที่กล่าวมาแล้ว) จึงจะสามารถใช้งานได้และเมื่อใช้คำสั่งนี้แล้วไม่ควรใช้คำสั่ง ON SELECTION PAD ซ้ำอีกเพราะจะทำให้เกิดการผิดพลาด (error) ได้ ส่วนการใช้งานซิสเต็มส์เมนูที่สั่งแอคตีฟนั้นจะสามารถใช้คีย์บอร์ดหรือเมาส์เหมือนกับการใช้งานเมนูที่ใช้คำสั่งอื่น ๆ ที่กล่าวมาแล้วสร้าง

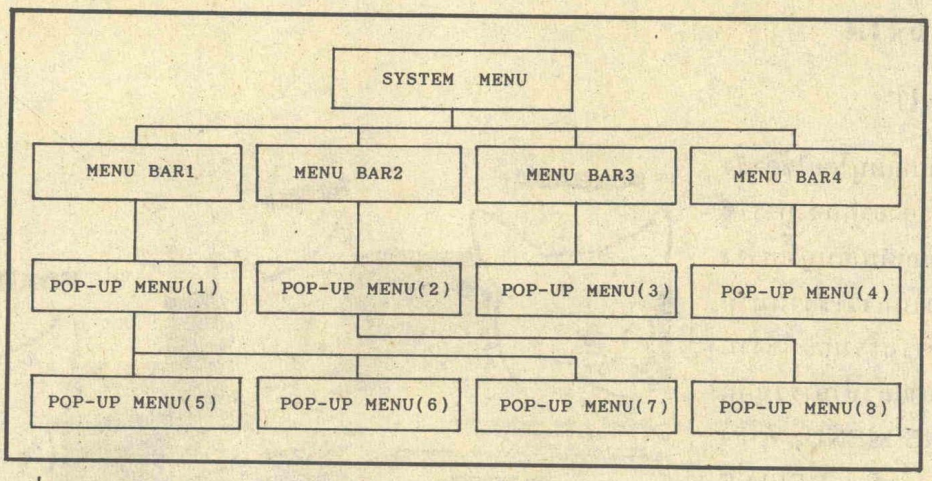
<pad menu> คือชื่อของหัวข้อเมนูบาร์ที่สร้างขึ้นจากคำสั่ง DEFINE PAD

<menu name> คือชื่อของเมนูที่สร้างขึ้นจากคำสั่ง DEFINE MENU

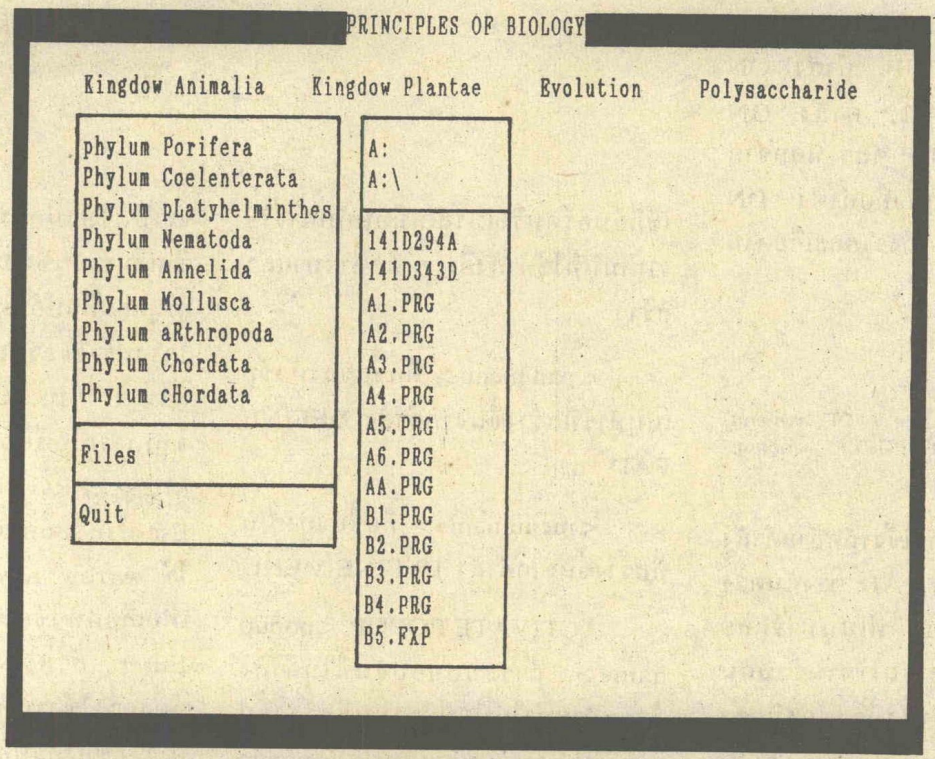
ACTIVATE POPUP <popup name> ถ้าระบบออปชันนี้ในคำสั่งด้วย จะทำให้ป๊อปอัพเมนูที่สร้างขึ้นแอคตีฟใช้งานร่วมกับเมนูบาร์ <popup name> คือชื่อของป๊อปอัพเมนูที่สร้างขึ้นจากคำสั่ง DEFINE POPUP แต่ถ้าไม่ระบุออปชันนี้ในคำสั่งจะทำให้เมนูบาร์แอคตีฟใช้งานเมนูเดียว (เหมือนกับการใช้คำสั่ง ON SELECTION PAD)

ตัวอย่างการสร้างและใช้งาน

โปรแกรม UPMMENU.PRG จะเป็นโปรแกรมตัวอย่างการใช้งานซิสเต็มส์เมนูซึ่งประกอบด้วยเมนูบาร์ 4 เมนูและป๊อปอัพเมนู 8 เมนู ซึ่งจะมีโครงสร้างของซิสเต็มส์เมนูดังรูปที่ 9 ส่วนรูปที่ 10 จะเป็นซิสเต็มส์เมนูที่ได้จากการเอ็กซีคิวต์โปรแกรม UPM-MENU.PRG การใช้งานเมนูจะใช้คีย์ Esc สำหรับยกเลิกการทำงานของเมนู, ใช้ Arrow Key สำหรับเลื่อนไฮไลต์เพื่อเลือกหัวข้อในเมนูใช้งานและใช้คีย์ Enter สำหรับเรียกหัวข้อของเมนูที่ต้องการใช้งานหรือจะใช้เมาส์แทนคีย์ต่าง ๆ ของคีย์บอร์ดก็ได้ จากโปรแกรม UPMMENU.PRG ให้ผู้อ่านทดลองใช้งานและศึกษาการทำงานของโปรแกรมดู จะทำให้เข้าใจและสามารถสร้างเมนูขึ้นใช้งานเองได้และให้ทดลองเปลี่ยนคำสั่ง ON PAD ในโปรแกรมเป็นคำสั่ง ON SELECTION PAD และสั่งเอ็กซีคิวต์โปรแกรมใหม่ ผู้อ่านจะเห็นถึงความแตกต่างของการใช้งานของสองคำสั่งนี้อย่างชัดเจน



รูปที่ 9 แสดงโครงสร้างของ System Menu จากโปรแกรม UPMMENU.PRG



รูปที่ 10 แสดง System Menu จากการเอ็กซีคิวต์โปรแกรม UPMMENU.PRG

ส่งท้าย

เมื่อผู้อ่านอ่านมาถึงตรงนี้แล้ว คงจะมีความรู้สึกที่ดีขึ้นเกี่ยวกับการสร้างและใช้งานเมนูนะครับ ส่วนการดัดแปลงเปลี่ยนรูปแบบในการแสดงเมนูบนจอสกรีนหรือวินโดว์นั้นขึ้นอยู่กับตัวผู้อ่านเองว่าจะประยุกต์ดัดแปลงกันอย่างไร สำหรับตัวโปรแกรมทั้งหมดทางกองบรรณาธิการวารสารไมโครฯ ได้จัดทำเป็นซอฟต์เมล์บริการแล้วครับ สวัสดีครับ